SOURCE CODE

```
/*===============================================================*/
/* procedure:     dsm_rindex.sas         revision:      01       */
/* authored_by:   doug mcnair            modified_by:   __       */
/* created_date: 04-JAN-2003             modified_date: __       */
/*===============================================================*/
/* purpose:       data transform for adjusting health service    */
/*                   for variations in rurality distance or access */
/* environment:   H:\cerdsm\NHS\test                             */
/*===============================================================*/
/*                                                               */
/* Read in data file. Assign the name "pop" to catchment pop.    */
/* Assign the name "d_nhs" to distance to care category to be    */
/* distance-index adjusted.                                      */
/*                                                               */
/* Only the infile statement and the the input statement need to */
/* be modified.                                                  */
/*                                                               */
/* This is a locally normal index and assigns a quantitative     */
/* measure of rurality to each participant in a study. Data req'd */
/* to compute the index are minimal; only two variables need be  */
/* measured. Analysis effort required to compute the index is    */
/* negligible, and the tradeoff between data collection and data */
/* analysis is defensible. Data collection is generally more     */
/* expensive than data analysis so minimize that. The two required*/
/* variables, county population and distance to care location, are*/
/* optimally transformed to achieve normality and weighted for   */
/* validity. Measure of departure from normality is automatically */
/* obtained while constructing the index. Positive score reflects */
/* rural loc or urban loc with compromised access. Negative score */
/* reflects an urban residence with good access relative to the  */
/* group under study. A score of zero reflects average distance  */
/* or accessibility of health services, rural or otherwise.      */
/* Reliability and construct validity were examined using two    */
/* data sets.                                                    */
/*                                                               */
/* The choice of a distance measure to gauge the distance between */
/* two distributions, in this case Gaussians, is critical.       */
/* The Kullback-Leibler, Bhattacharyya and arc-cosine distances  */
/* are all related to maximum likelihood, and reduce to the      */
/* Mahalanobis distance when used to measure the distance between */
/* two Gaussian distributions with equal covariance matrices.    */
/* The advantage of the arc cosine distance is that it has a     */
/* closed form expression, implemented in IML.                   */
/*                                                               */
/*===============================================================*/

data distance;
     infile 'c:\0_cerdsm\health_econ\rurality\cancerpop.dat';
     input case_id d_nhs pop;
     retain d_nhs pop;
     use=1;
     if d_nhs = 0 then d_nhs=.5;
     if d_nhs =. then use=0;
     if pop = . then use=0;
     keep d_nhs pop use;
     if use = 1 then output;
  proc means;
  proc freq;
     tables d_nhs;

  proc iml;
    use distance; reset noname; read all var _num_ into YY;
    step=.05; mino=step; maxo=4.; guess={1, 1};
    k=2; c={.5, -1}; n=nrow(YY);
    do i=1 to k; YY[,i]=YY[,i]/max(YY[,i]); end;
    j=(1:n)`; p=(j-.5)/n; pi=arcos(-1);
    X=J(n,k,0);  X1=J(n,k,0);  X2=J(n,k,0);
    eps=1/2**26;


    start deriv;
    tau=lam-1; omega=(ssq(tau)); romega=sqrt(omega); IMP=I(k);
    if romega >= eps then do;
    tau=tau/romega; IMP=IMP-tau*tau`; end;
    do j=1 to k;
      y=YY[,j];
      if lam[j] =0 then do;
        t=log(y);
        t1=t#t/2;
        t2=2*t1#t/3;  end;
      else do;
        t=sign(lam[j])*y##lam[j];
        t1=t#log(y);  t2=t1#log(y); end;
      t=t-J(n,1,sum(t)/n); v0=ssq(t);  xj=t/sqrt(v0); X[,j]=xj;
      t1=t1-J(n,1,sum(t1)/n);  t2=t2-J(n,1,sum(t2)/n);
      v01=xj`*t1;  v02=xj`*t2;
      x1j=(t1-xj*v01)/sqrt(v0);  X1[,j]=x1j; v11=x1j`*t1;
      x2j=(t2-xj*v02-2*x1j*v01-xj*v11)/sqrt(v0); X2[,j]=x2j;
      end;
      z=X*c;  s2=z`*z/(n-1); s=sqrt(s2);  z=z/s;
      ii=rank(z);  B=z; z[ii]=B;
      B=X;  X[ii,]=B;  B=X1; X1[ii,]=B;  B=X2; X2[ii,]=B;
      G=sqrt(omega/s2)*diag(c)*(X1`-(X1`*z)*z`/(n-1));
      Phi=probnorm(z);
      Phi = Phi#(eps <= Phi)#(Phi <= 1-eps) + J(n,1,eps)#(Phi < eps)
          + J(n,1,1-eps)#(1-eps < Phi);
      logphi=log(Phi); logphi1=log(1-Phi);
      ph=exp(-z#z/2)/sqrt(2*pi);  logph=-(z#z+J(n,1,log(2*pi)))/2;
      A2n=2*p`*logphi1-2*p`*logphi-2*sum(logphi1)-n;
      a1=-2*exp(logph-logphi-logphi1)#(p-Phi);
      a2=2*exp(2*logph-2*logphi-2*logphi1)#((p-Phi)#(p-Phi)+p#(1-p));
      a2=a2+2*exp(logph-logphi-logphi1)#z#(p-Phi);
      d1=G*a1; f=IMP*d1;
      d2=(omega/s)*diag(c)*(X2`*a1-(X2`*z)*(z`*a1)/(n-1));
      d3=sqrt(omega/s2)*IMP*diag(c)*X2`*z;
      H=-(f*tau`+tau*f`);  H1=diag(d2)-(tau`*d1)*I(k)-(d1*d3`+d3*d1`)/(n-
1);
      H1=H1-G*G`*(z`*a1)/(n-1)+(G#(J(k,1,1)*a2`))*G`;
      H=H+IMP*H1*IMP;
      finish;

    start optimize;
```

```
 delta=1;  A2n0=n;
 do while (delta > 1.e-5);
 run deriv;
 call gsorth(T,u,lindep,I(2)-tau*tau`);  T=T[,1];
 b=-inv(T`*H*T)*(T`*f);
 tau=tau+T*b;
 tau=tau/sqrt(ssq(tau));
 lam=1+tau*romega;
 delta=n*abs(A2n0-A2n);
 A2n0=A2n;
 end;
 finish;


start search;
filename out 'power.out';
file out;
lam={1,1}; run deriv;
put @2 romega 7.4 +3 A2n 9.4+3 (lam[1]) 7.4+3 (lam[2]) 7.4+3;
lam=guess-.001;
do romega=mino to maxo by step;
omega=romega**2;
tau=lam-1;  tau=tau/sqrt(ssq(tau));  lam=1+romega*tau;
run optimize;
put @2 romega 7.4 +3 A2n 9.4+3 (lam[1]) 7.4+3 (lam[2]) 7.4+3;
end;
closefile out;
finish;

lam={0.2,-.1}; run deriv; print lam A2n;
lam={0,-.15}; run deriv; print lam A2n;
lam={0,-.2}; run deriv; print lam A2n;
lam={1.,1.}; run deriv; print lam A2n;
lam={0.,0.}; run deriv; print lam A2n;
run search;
```

SOURCE CODE

```
/*==================================================================*/
 /* procedure:    dsm_riskadj.sas        revision:      01          */
 /* authored_by:  doug mcnair            modified_by:   __          */
 /* created_date: 04-JAN-2003            modified_date: __          */
 /*=================================================================*/
 /* purpose:      calculate risk-adjusted area rates for indicators*/
 /*                  variations in rurality distance or access      */
 /*                  adj for age and dcat                           */
 /* environment:  H:\cerdsm\NHS\test                                */
 /*=================================================================*/
 /*                                                                 */
 filename control 'h:\cerdsm\nhs\control.sas';

 %include control;

 %macro area;
   %if &msalevl eq 1 %then msa;
      %else hospstco;
 %mend;

 %macro cty2msa;
    %if &msalevl eq 1 %then %do;
       label msa ='msa %area assigned from hospstco';
       msa = input( put(hospstco,area.), 5.);
    %end;
 %mend;

 %macro sum2msa;
    %if &msalevl eq 1 %then %do;
        proc summary nway;
            class msa popcat dcat;
            var pop;
            output out=qipop (drop=_type_ _freq_) sum=;
        run;
    %end;
 %mend;

 /* ----------------------------------------------------------------*/
 /*                      add pop denominator                        */
 /* ----------------------------------------------------------------*/

 proc   sort data=in3.&infilea3. (keep=%area) out=%area nodupkey;
 by     %area;
 run;


 data   qipop;
 length hospstco 8 popcat 3 dcat $1 ;
 infile popfile3 missover;
 input  hospstco popcat dcat pop;
 %cty2msa
 run;

 %sum2msa
 proc    sort data=qipop;
 by      %area popcat dcat;
 run;

 data    qipop;
 merge   %area(in=x) qipop;
 by      %area;
 if x;
 run;

 /* --------------------------------------------------------------*/
 /* select obs for each indicator from previous step in turn.      */
 /* each iteration passes vars that control covars for that indic: */
 /*     n - obs number from covar files                            */
 /*         also identifies format used to index covars            */
 /*     pq - indicator name without prefix                         */
 /*     a - number of demographic categories (age and d_nhs)       */
 /*     s - determines whether age*d interactions are needed       */
 /*     o - determines omitted age cat in format                   */
 /* --------------------------------------------------------------*/

 %macro mod3(n,pq,a,s,o);

 /* --------------------------------------------------------------*/
 /* SET creates TEMP1, containing the dep var TPQ and indep vars   */
 /* used in the regression.  append to enctr data one obs per %area */
 /* and demographic grp                                            */
 /* --------------------------------------------------------------*/

 data   temp1(keep=key %area t&pq. d age1-age%eval(2*&a.) aidx pop
mergevar);
 length age1-age%eval(2*&a.) aidx 8;
 set    in3.&infilea3.(keep=key %area t&pq. dcat agecat age)
        qipop;

 /* --------------------------------------------------------------*/
 /* retain only enctr/area/demographic grp needed for this indicator*/
 /* --------------------------------------------------------------*/

 n = &n.;

 if %area eq . then delete;

 if pop > 0 and n notin (2,9) then t&pq. = 0;

 if agecat in ('1') or popcat in (1,2,3,4) then do;
   if n in (1,3,5,7,8,13,14,15,16)
     then t&pq. = .;
 end;
 else if agecat in ('2') or popcat in (5,6,7,8) then do;
   if n in (4,6,9)
     then t&pq. = .;
 end;
 else if agecat in ('3','4') or
         popcat in (9,10,11,12,13,14,15,16,17,18) then do;
   if n in (4,6,9)
     then t&pq. = .;
```

```
end;

if t&pq. ne .;

/* ----------------------------------------------------------------*/
/* initialize covar dummies to zero                                */
/* ----------------------------------------------------------------*/

array agex{%eval(2*&a.)} age1-age%eval(2*&a.);

d = 0;
do i = 1 to %eval(2*&a.);
  agex(i)=0;
end;

/* ----------------------------------------------------------------*/
/* set index for age array (AIDX). categories vary by indicator and*/
/* are determined by format corresponding to N                     */
/* ----------------------------------------------------------------*/

if age > 0 then aidx = put(age,a&n.fmt.);
else if pop > 0 then do;
  aidx = popcat;
  if aidx = &o. then aidx = 0;
end;
else aidx = 0;

/* ----------------------------------------------------------------*/
/* set appropriate covariate dummy to 1 for this enctr             */
/* ----------------------------------------------------------------*/

if dcat in (&s.) then d = 1;
if aidx > 0 then do; agex(aidx) = 1; agex(&a.+aidx) = 1 * d; end;

mergevar = 1;

run;

proc   sort data=temp1;
by %area d aidx pop;
run;

/* ----------------------------------------------------------------*/
/* subtract number of encounters from %area/demographic group pop  */
/* and set weight of 1 for each obs from enctr data, and a         */
/* weight of (pop-encounters) for the appended obs for each group  */
/* ----------------------------------------------------------------*/
data temp1(keep=key %area t&pq. d age1-age%eval(2*&a.) wht mergevar);
set  temp1;
by %area d aidx;

retain wcnt 0;

if first.aidx then wcnt = 0;
if pop > 0 then wht = pop - wcnt;
else wht = 1;
wcnt + 1;
```

```
run;

/* --------------------------------------------------------------*/
/* read overall mean file for indicator; F variables are filler   */
/* --------------------------------------------------------------*/

data   temp1m(drop=n f1-f%eval(40-(2*&a.)));
length n f1-f%eval(40-(2*&a.)) 3 md mage1-mage%eval(2*&a.) 8;
infile meansa missover lrecl=1000;
input n md mage1-mage%eval(2*&a.) f1-f%eval(40-(2*&a.));
mergevar = 1;
if n = &n.;

run;

/* --------------------------------------------------------------*/
/* subtract overall mean of vars from the vars on each obs       */
/* so that adj area rate is the rate for the average person      */
/* --------------------------------------------------------------*/

data   temp1(keep=key %area t&pq. d age1-age%eval(2*&a.) wht);
merge  temp1 temp1m;
by     mergevar;

array  covar{%eval(1+(2*&a.))}  d  age1-age%eval(2*&a.);
array  means{%eval(1+(2*&a.))} md mage1-mage%eval(2*&a.);

do i = 1 to %eval(1+(2*&a.));
  covar(i) = covar(i) - means(i);
end;

run;

/* --------------------------------------------------------------*/
/*            read regression coeffs for each covariate          */
/* --------------------------------------------------------------*/

data   temp2(keep=intercep d age1-age%eval(2*&a.) _type_ _model_
            _depvar_ t&pq.);
length intercep d age1-age%eval(2*&a.) f1-f%eval(40-(2*&a.)) t&pq. 8
       _type_ _model_ _depvar_ $8;
infile covara missover lrecl=1000;
input  n intercep d age1-age%eval(2*&a.) f1-f%eval(40-(2*&a.))
       _depvar_ _model_ _type_ _rmse_ t&pq.;

if n = &n.;

run;

/* --------------------------------------------------------------*/
/* calculate predicted indicator for each obs using actual covars */
/* and the estimated coeffs, on de-meaned sample, where %area     */
/* effects are removed by subtracting the predicted val from the  */
/* actual on each obs, and calculating the mean difference for the */
/* %area.  See Green WH. Econometric Analysis 2e. 1990 pp. 466-9.  */
/* --------------------------------------------------------------*/
```

```
proc   score data=temp1 score=temp2 type=parms out=temply;
var    d age1-age%eval(2*&a.);
run;

data   temply;
set    temply;

ehat = t&pq. - mhat;
one = 1;

run;

proc   summary data=temply ;
class  %area;
var    ehat one;
output out=r&pq. mean(ehat)=r&pq. sum(one)=p&pq.;
weight wht;
run;

%mend;

%MOD3(1,APQ01,18,'F',6);
%MOD3(2,APQ02,4,'F',0);
%MOD3(3,APQ03,18,'F',10);
%MOD3(4,APQ04,18,'F',1);
%MOD3(5,APQ05,18,'F',12);
%MOD3(6,APQ06,18,'F',1);
%MOD3(7,APQ07,18,'F',10);
%MOD3(8,APQ08,18,'F',13);
%MOD3(9,APQ09,2,'F',0);
%MOD3(10,APQ10,18,'F',1);
%MOD3(11,APQ11,18,'F',5);
%MOD3(12,APQ12,18,'F',6);
%MOD3(13,APQ13,18,'F',11);
%MOD3(14,APQ14,18,'F',8);
%MOD3(15,APQ15,18,'F',7);
%MOD3(16,APQ16,18,'F',11);

/* --------------------------------------------------------------------*/
/*            merge %area adjusted rates for indicators                */
/* --------------------------------------------------------------------*/

data   riskadj;
length %area RAPQ01-RAPQ16 PAPQ01-PAPQ16 8;
merge  RAPQ01(KEEP=%AREA RAPQ01 PAPQ01)
       RAPQ02(KEEP=%AREA RAPQ02 PAPQ02)
       RAPQ03(KEEP=%AREA RAPQ03 PAPQ03)
       RAPQ04(KEEP=%AREA RAPQ04 PAPQ04)
       RAPQ05(KEEP=%AREA RAPQ05 PAPQ05)
       RAPQ06(KEEP=%AREA RAPQ06 PAPQ06)
       RAPQ07(KEEP=%AREA RAPQ07 PAPQ07)
       RAPQ08(KEEP=%AREA RAPQ08 PAPQ08)
       RAPQ09(KEEP=%AREA RAPQ09 PAPQ09)
       RAPQ10(KEEP=%AREA RAPQ10 PAPQ10)
       RAPQ11(KEEP=%AREA RAPQ11 PAPQ11)
       RAPQ12(KEEP=%AREA RAPQ12 PAPQ12)
       RAPQ13(KEEP=%AREA RAPQ13 PAPQ13)
       RAPQ14(KEEP=%AREA RAPQ14 PAPQ14)
       RAPQ15(KEEP=%AREA RAPQ15 PAPQ15)
       RAPQ16(KEEP=%AREA RAPQ16 PAPQ16);
by     %area;

label
RAPQ01 = 'DIABETES SHORT TRM COMPLICATN (Risk Adj)'
RAPQ02 = 'PERFORATED APPENDIX           (Risk Adj)'
RAPQ03 = 'DIABETES LONG TERM COMPLICATN (Risk Adj)'
RAPQ04 = 'PEDIATRIC ASTHMA              (Risk Adj)'
RAPQ05 = 'COPD                          (Risk Adj)'
RAPQ06 = 'PEDIATRIC GASTROENTERITIS     (Risk Adj)'
RAPQ07 = 'HYPERTENSION                  (Risk Adj)'
RAPQ08 = 'CONGESTIVE HEART FAILURE      (Risk Adj)'
RAPQ09 = 'LOW BIRTH WEIGHT              (Risk Adj)'
RAPQ10 = 'DEHYDRATION                   (Risk Adj)'
RAPQ11 = 'BACTERIAL PNEUMONIA           (Risk Adj)'
RAPQ12 = 'URINARY INFECTION             (Risk Adj)'
RAPQ13 = 'NEW CANCER MGT DELAY>14D S/P  (Risk Adj)'
RAPQ14 = 'DIABETES UNCONTROLLED         (Risk Adj)'
RAPQ15 = 'ADULT ASTHMA                  (Risk Adj)'
RAPQ16 = 'LOWER EXTREMITY AMPUTATION    (Risk Adj)'
;

label
PAPQ01 = 'DIABETES SHORT TRM CMPLICTN (Denominator)'
PAPQ02 = 'PERFORATED APPENDIX         (Denominator)'
PAPQ03 = 'DIABETES LONG TRM CMPLICTN  (Denominator)'
PAPQ04 = 'PEDIATRIC ASTHMA            (Denominator)'
PAPQ05 = 'COPD                        (Denominator)'
PAPQ06 = 'PEDIATRIC GASTROENTERITIS   (Denominator)'
PAPQ07 = 'HYPERTENSION                (Denominator)'
PAPQ08 = 'CONGESTIVE HEART FAILURE    (Denominator)'
PAPQ09 = 'LOW BIRTH WEIGHT            (Denominator)'
PAPQ10 = 'DEHYDRATION                 (Denominator)'
PAPQ11 = 'BACTERIAL PNEUMONIA         (Denominator)'
PAPQ12 = 'URINARY INFECTION           (Denominator)'
PAPQ13 = 'NEW CANCER MGT DELAY>14D S/P(Denominator)'
PAPQ14 = 'DIABETES UNCONTROLLED       (Denominator)'
PAPQ15 = 'ADULT ASTHMA                (Denominator)'
PAPQ16 = 'LOWER EXTREMITY AMPUTATION  (Denominator)'
       ;

array arry1{16} rapq01--rapq16;
do i = 1 to 16;
  if arry1(i) < 0 then arry1(i) = 0;
  else if arry1(i) > 1 then arry1(i) = 1;
end;

drop i;

run;

/* -----------------------------------------------------------------*/
/* smoothed demeaned rates = area rate * (signal / signal + noise) */
/* -----------------------------------------------------------------*/
```

```
 data   smoothed(DROP=RAPQ01-RAPQ16);
 set    riskadj;

 %include msx;

 array arrya{16} RAPQ01-RAPQ16;
 array arryn{16} PAPQ01-PAPQ16;
 array arryx{16} SAPQ01-SAPQ16;

 do i = 1 to 16;
   arryx(i) = arrya(i) - arrya3(i);
   if arryn(i) ne 0 and arrya2(i) ne 0 then
      arryx(i) = arryx(i) * (arrya2(i) / (arrya2(i) + (arrya1(i) /
arryn(i))));
   else arryx(i)=.;
   arryx(i) = arryx(i) + arrya3(i);
 end;

 label
 SAPQ01 = 'DIABETES SHORT TRM COMPLICATN (Smoothed)'
 SAPQ02 = 'PERFORATED APPENDIX          (Smoothed)'
 SAPQ03 = 'DIABETES LONG TERM COMPLICATN (Smoothed)'
 SAPQ04 = 'PEDIATRIC ASTHMA             (Smoothed)'
 SAPQ05 = 'COPD                         (Smoothed)'
 SAPQ06 = 'PEDIATRIC GASTROENTERITIS    (Smoothed)'
 SAPQ07 = 'HYPERTENSION                 (Smoothed)'
 SAPQ08 = 'CONGESTIVE HEART FAILURE     (Smoothed)'
 SAPQ09 = 'LOW BIRTH WEIGHT             (Smoothed)'
 SAPQ10 = 'DEHYDRATION                  (Smoothed)'
 SAPQ11 = 'BACTERIAL PNEUMONIA          (Smoothed)'
 SAPQ12 = 'URINARY INFECTION            (Smoothed)'
 SAPQ13 = 'NEW CANCER MGT DELAY>14D S/P (Smoothed)'
 SAPQ14 = 'DIABETES UNCONTROLLED        (Smoothed)'
 SAPQ15 = 'ADULT ASTHMA                 (Smoothed)'
 SAPQ16 = 'LOWER EXTREMITY AMPUTATION   (Smoothed)'
 ;

 drop i;

 run;

 /* ---------------------------------------------------------------*/
 /*                          merge rates                           */
 /* ---------------------------------------------------------------*/

 data   out3.&outfila3.;
 merge  in3.&infilea2. (where=(_type_ in(0,8))
     keep=%area _type_ tapq01-tapq16 papq01-papq16 oapq01-oapq16)
          riskadj(keep=%area rapq01-rapq16)
          smoothed(keep=%area sapq01-sapq16) ;
 by     %area;
 run;

 /* ---------------------------------------------------------------*/
 /*                          output rates                          */
 /* ---------------------------------------------------------------*/
```

```
proc means data=out3.&outfila3.(where=(_type_ in (8))) n nmiss min max
      mean sum;
 title4  'summary of area-level rates (_type_=8)';
 run;
```

CANCER POP. DATA

100	35	12083
101	20	11697
102	15	5046
103	30	59218
104	20	9505
105	25	113419
106	20	8669
107	90	12083
108	10	1383
109	10	8669
110	5	14562
111	10	9505
112	45	8669
113	15	12083
114	25	8239
150	20	113419
151	45	59218
152	8	5452
153	45	1503
154	120	25010
155	15	11337
156	5	6536
157	5	9505
158	5	5163
159	25	8669
160	45	78687
161	30	6536
162	5	10505
163	15	4106
164	40	12083